

Unified SDLC Platform

Every Role, Every Phase - One Intelligent Workspace



Legacy



Startup



Code Analysis



Documentation



Diagram



WireFrame



Modern Engineering Challenge — and a Smarter Way Forward

Software delivery today is **fast, fragmented, and heavily tool-dependent**.

Teams struggle to connect business intent, design, documentation, and code **in one flow**.

Decisions get delayed, quality varies, and scaling adds **cost instead of value**.



- ✓ A **single workspace** to define, design, and deliver projects.
- ✓ **Interlinked** modules for documentation, data, API, and code generation.
- ✓ Real-time collaboration — **decisions finalized in the meeting itself**.
- ✓ Built-in validation ensuring **every change stays consistent**.
- ✓ **Reusable templates** and enterprise assets for **faster, reliable** delivery.

LensHub – One View. All Answer. No Doubt.

	Analysis	Upgrade	Migration
I want to build knowledge base (know what I have)	#1 – Discovery #2 – Modernize		
I want to move to latest version (upgrade code)		I help	
I want to switch technology (from old to new)	Will do first		Fresh Code
Code Analysis	Needed	Partial	Partial
Code Generation		Single Batch	By feature
AI Provider (Public, Private), Custom Prompt Bank	Inbuilt + Design Your Own Prompt		

Decode > Align > Deliver

Built for the Gaps That Slow You Down

CODE INFORMATION

Frontend, backend, dependency view
Extract inputs & actions

01

DATABASE INFORMATION

Table, Views, Procedure, Functions
Column, Parameters, Data types

02

CONTEXTUAL CODE SNIPPET ACCESS

Handy code stub/snippet view that is
mapped with UI or code layer

03

VISUAL REPRESENTATION - DIAGRAMS

Dependency View, Diagrams - Class, Flow,
Use Case, Mind map, Swimlane, BPM

04



Hidden Gems
&
Outcome



INPUT SCHEMA, USER ACTIONS & BEHAVIOUR

Micro requirement - input fields, events,
user actions, validations, basic wire framing

05

ACCEPTANCE & TEST DATA MANAGEMENT

NFR, Define acceptance criteria & test data
for positive & negative testing

06

COLLABORATION & PROJECT MANAGEMENT

Basic team collaboration like contextual
chat, task and release planning

07

BOILERPLAT CODE GENERATION

Code generation as per architect and design
pattern for choice of technologies

08

Product Owner | Business Analyst | UI Design | Developer | QA | Project Manager | Technical Architect

Benefits of LensHub

- LensHub **evaluates** application **within short time** and generate reports & technical and functional documentation
- **Inter dependency visuals** eliminates spending countless hours of tech team
- Easy and **advance business requirement** documentation, with **diagram and whiteboard**
- Realtime **wireframe** for your **business forms**
- **Saves more than 40% of team's time** and results into significant cost saving throughout the digital transformation journey
- Your **choice of technical decision** on modernization approach – technology, cloud etc.
- An inbuilt **application life cycle management** features to eliminate multiple systems need
- **Generates** multi layer **code** in your choice of technology for modern application development



What to do next? – Where to take first modernization step?

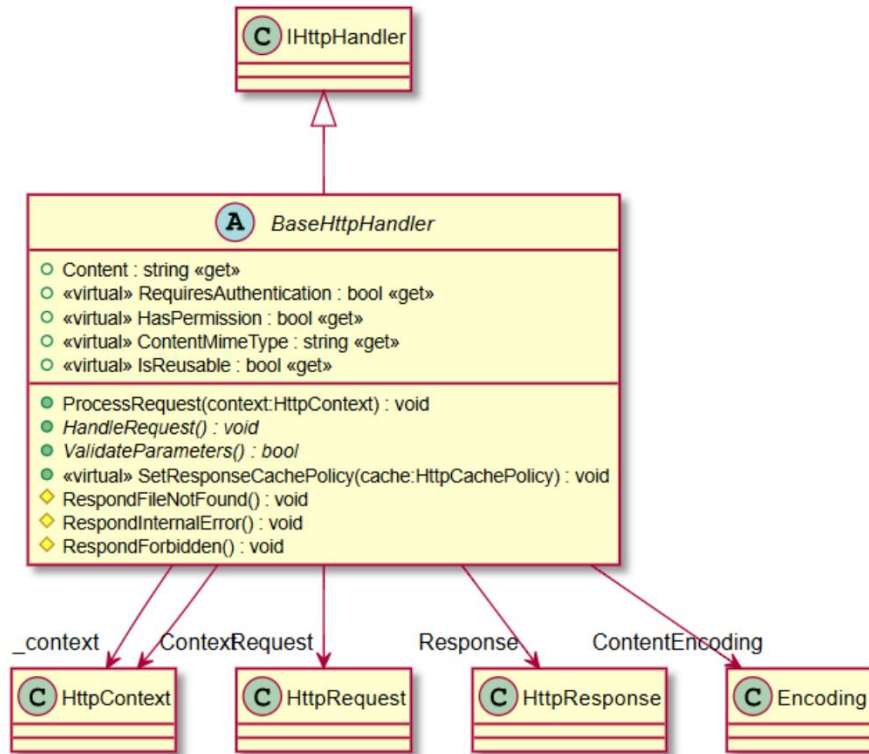


I want to take
glimpse



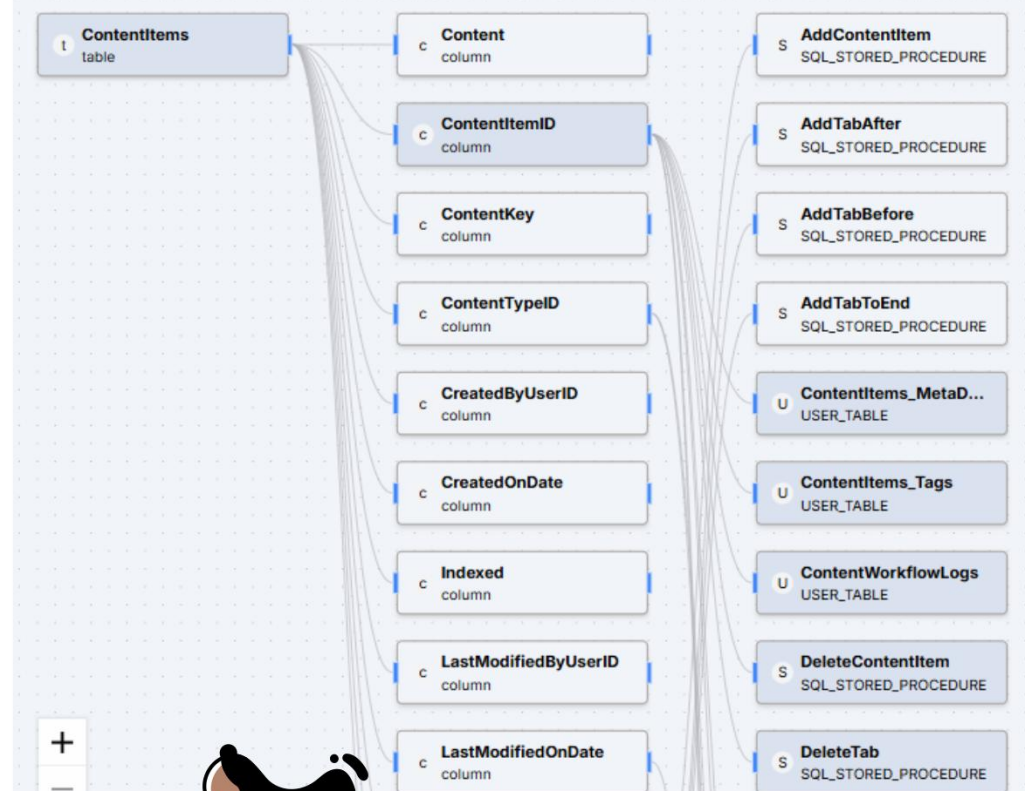
BaseHandler.cs

Diagram

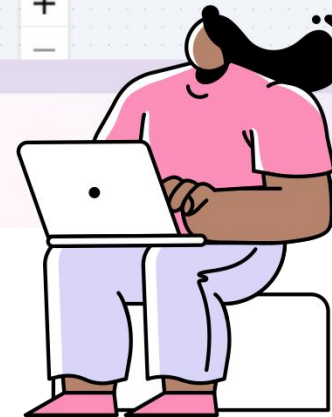


ContentItems

Dependency



Code Analysis



Master Layout Concept UI

Home Create Analytics

Home > Create Blog

Title

Category

Format

☐ Html ☐ Markdown

Content

E-commerce Checkout Process

```
@startuml
    title E-commerce Checkout Process Sequence
    actor Customer
    participant "Product Page" as ProductPage
    participant "Shopping Cart" as Cart
    participant "Order System" as OrderSystem
    participant "Payment Gateway" as PaymentGateway

    Customer -> ProductPage: View Products
    Customer -> ProductPage: Add Product to Cart
    ProductPage -> Cart: Add Item
    Customer -> Cart: View Cart
    Customer -> OrderSystem: Place Order
    OrderSystem -> PaymentGateway: Process Payment
    PaymentGateway -> OrderSystem: Payment Confirmed
    OrderSystem -> Customer: Order Confirmation
@enduml
```

E-commerce Checkout Process Sequence

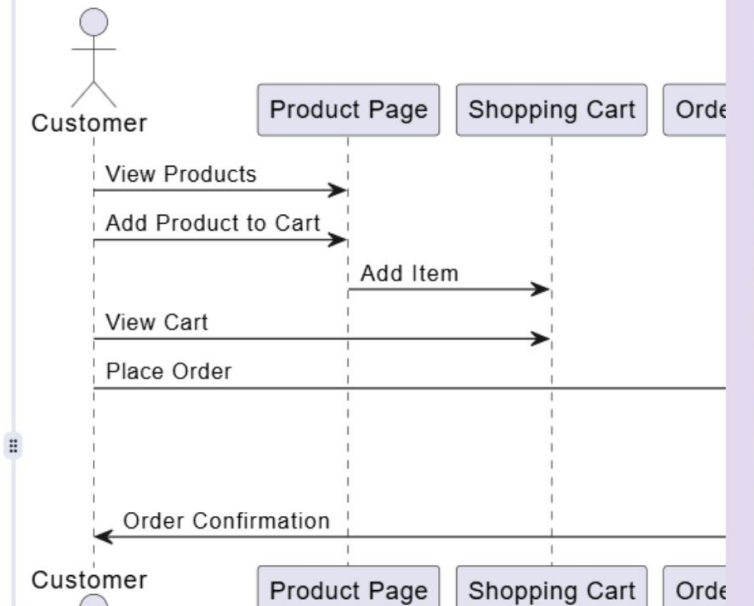
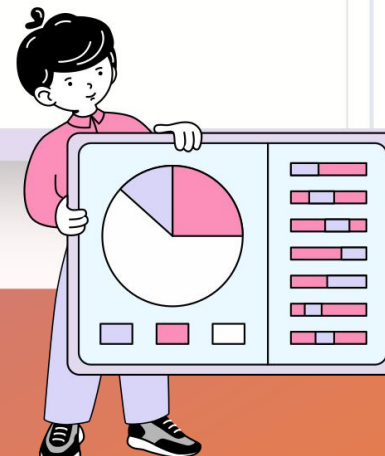


Diagram | Wireframe



Documentation



DotNetNuke 9



Create

Blog Module

Back

Export

Dependency

Audit Trail

Submit

Save As Template

Introduction



Blog Entry Page



Blog List Page



The Blog Entry Page in a DotNetNuke (DNN) website is designed to display individual blog posts along with relevant information such as author details, categories, tags, comments, and related content. This document outlines the requirements for the Blog Entry Page, focusing on the inclusion of multiple authors and the associated business logic. The goal is to ensure a seamless and engaging user experience, as well as to support the management and display of content created by multiple authors.

1. Author Details and Multiple Authors

Prompt

Write user story for a sign-up with GitHub account for a subscriber user role.

AI Engine Provider Name

OpenAI

e there are

Dependency

Signup - By Agency
Alm Business Form

User Feature - Core T...
Alm Test Scenario

Categories & Sections
Alm Test Suite

Categories & Sections
Alm Test Planner



Documentation

Business Form

Search Anything...

DotNetNuke 9

Add

Action

Mockup

Schema

Rules

Event

Signup - By Agency

DEFAULT

email

last name

firstname

User Information

User Name

Password

Profile Information

Profile Photo

BA

Field Name

User Name

Field Type Name

TEXT

Mandatory Insert Remarks

Please enter valid email for user name

Mandatory Update Remarks

You cannot update the user name after created.

Upload Type

FRONTEND

Framework Control Name

TEXTBOX

Frontend Display Control Name

TEXTBOX

Valid Extensions

Custom Option

Validation Message

Please enter user name as email for

Validation Type

TEXT

BACKEND

Min Text Length

3

Max Text Length

50

Min Number Range

Max Number Range

Min Date Value

Max Date Value

DB

Table Name

users

Column Name

user_name

Column Relation

Please select

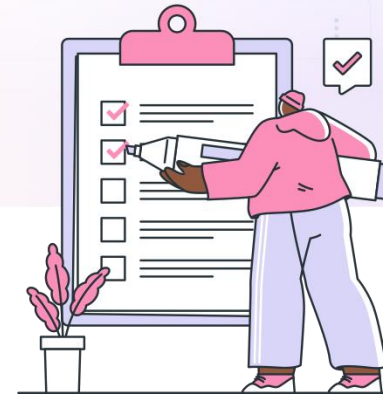
Column Data Type

varchar(255)

Remarks for DB Team

Store data in lower case, the data value must be unique, add index as this field is going to be searchable by users.

Micro Requirement



Customer Dashboard

Customer Dashboard

Open
4

WIP
21

Hold
0

Finished
4

Customer Support Dashboard

Customer Support Dashboard

Open
5

WIP
23

Hold
0

Finished
0

Finance Dashboard

Finance Dashboard

Open
3

WIP
28

Hold
0

Finished
3

Inventory

Inventory M

Open
1

Board View

Search Anything...

DotNetNuke 9

August Release

← Back

Export

Import

Open- 2

WIP- 18

Hold- 0

Finished- 2

Columns

Ready for Testing 2

Testing In Progress 3

Awaiting Information 3

Deferred 2

Reopened 5

Dashboard personalization features

Release

Demo1 User2

Inconsistent UI on Mobile

Release

Demo1 User1

Duplicate Accounts Allowed

Release

Demo1 User1

Cross-Site Scripting (XSS) Vulnerability

Release

Demo1 User1

Analytics Not Tracking

Release

dem

Customer onboarding improvements

Release

Demo1 User1

Chat

Search Anything...

DotNetNuke 9

Group Chat

Backlog Team

DU

Hi Sarah,
12/16/2024 23:44

Hi John, can you ch

Collaboration



Search Anything...

DotNetNuke 9

Back

Export

Submit

Delete

Testing Scenarios of Sign-Up by Agency

Title

Session Timeout During Sign-Up

Test Case Type

Acceptance Testing

Test Case Sub Type

Business Form/Documentation

Please select

Short Description

Validate the behavior when the session times out during the sign-up process

Pre Condition Note

Session timeout is configured

Post Condition

Labels

Press enter to add in list

+ Test Step

Api Testing

Search A

Create

Signup API - For Inhouse Application

Signup

Signup API - For Inh...

POST

{api-application-url}/signup/create

Params

Authorization

Headers

Cookies

Req. Body

Res. Body

1

2

3

"user_name": "lenshub",

"email" : "hello@languageloop.io"

Test Management



Language

☒ TypeScript ☐ Javascript ☐ Dart

Framework

☐ Angular ☐ React (With Typescript) ☐ Vue Js (With Typescript) ☐ Next Js (With Typescript) ☐ Nuxt Js (With Typescript)
☐ Svelte (With Typescript) ☐ Stencil ☐ React Js ☐ Angular (Javascript) ☐ js
☐ Next Js (Javascript) ☐ Nuxt Js (Javascript) ☐ Ember Js ☐ Backbone Js ☐ Preact
☐ Alpine Js ☐ Svelte ☐ Flutter (Web) ☐ Angulardart ☐ Overreact
☐ Sapper (With Dart)

Api Architecture

☐ Dio
☐ GraphQL Client Libra
☐ React Query (Now 1
Query)
☐ Superagent
☐ Swr

New Session

User Table Script >
 User Profile Code >
 Shipping Address Script >
 Order Form Code

Order Form Code

Code with module, service as application (B)

```

import { useState } from 'react';
import { useForm } from 'react-hook-form';
import { useMutation } from 'react-query';

interface OrderForm {
  orderId: string;
  customerName: string;
  orderDate: string;
  customerEmail: string;
  notes: string;
  totalAmount: number;
  trackingNumber: string;
  shippingAddress: string;
  orderStatus: string;
  paymentMethod: string;
  productDetails: string;
}

const OrderFormComponent = () => {
  const { register, handleSubmit, formState: { errors } } = useForm<OrderForm>();
  const [characterCount, setCharacterCount] = useState(0);
  const [StatusOptions, setOrderStatusOptions] = useState<{ value: string; label: string }[]>([]);
  const [MethodOptions, setPaymentMethodOptions] = useState<{ value: string; label: string }[]>([]);
  const [picker, setShowDatePicker] = useState(false);

  const addOrder = useMutation('addOrder', async (data: OrderForm) => {
    // ...
  });
    
```

Code Generation





hello@lenshub.io
